REMARKS

I. Introduction

In response to the Office Action dated January 4, 2006, no claims have been cancelled, amended or added. Claims 1, 3-10, 13-21 and 23-30 remain in the application. Re-examination and re-consideration of the application is requested.

II. Double Patenting Rejection

On page (2) of the Office Action, claims 1 and 3-30 were provisionally rejected on the ground of nonstatutory double patenting over claim 1-33 of copending Application No. 10/807,871.

Applicant's attorney notes the provisional nature of these rejections, and will respond substantively to the rejections once allowable claims have been identified.

III. Prior Art Rejections

A. The Office Action Rejections

On page (3) of the Office Action, claims 1 and 3-30 were rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 6,434,545 (MacLeod) in view of U.S. Patent No. 6,496,819 (Bello).

Applicant's attorney respectfully traverses these rejections.

B. The Applicant's Claimed Invention

Applicant's claimed invention, as recited in independent claims 1, 11, and 21, is generally directed to a method of optimizing execution of a query that accesses data stored on a data store connected to a computer. Claim 1 is representative and recites the steps of generating cardinality estimates for one or more query execution plans for the query using statistics of one or more automatic summary tables that vertically overlap the query, and using the generated cardinality estimates to determine an optimal query execution plan for the query.

C. The MacLeod Reference

MacLeod describes a user specifying one or more queries comprising a batch of SQL statements. Each query submitted is displayed and represented as a tree, with each operation in the execution plan for the query represented by a corresponding tree node. This representation intuitively conveys the DBMS execution strategy which would be used to process the queries. The

tree nodes are displayed as icons, with a unique one of such icons corresponding respectively to each of the possible query operations. In addition, the computing cost of each operation (each node) as a percentage of overall query cost is displayed, as is the cost of each query as a percentage of the overall cost of the specified query batch. A user may select an operation (tree node) with a conventional mouse, whereupon a user interface will show more detailed cost statistics relating to the selected operation.

D. The Bello Reference

Bello describes a method and system for processing queries. Specifically, techniques are provided for handling a query that does not reference a particular materialized view, where the query requires access to values from a particular column not contained in the materialized view. A technique is also provided for processing a query that does not reference a particular materialized aggregate view, where the materialized aggregate view specifies an outer join between a child table and a parent table and the query specifies a particular type of join between the child table and the parent table, where the particular type of join is one of an inner join, an anti-join and a semi-join. The query is rewritten to produce a rewritten query that accesses the materialized aggregate view to produce data required by the query. A technique is also provided for processing a query that does not reference a particular materialized view and that specifies that results are to be grouped by a first set of one or more columns, where the materialized view reflects data that is grouped by a second set of one or more columns. If each column in the first set of columns either exactly matches a column in the second set of columns, is functionally dependent on another column in the first set of columns, or is functionally dependent on a column in the second set of columns, then the query is rewritten to produce a rewritten query that references the materialized view.

E. Applicant's Claimed Invention Is Patentable Over The Cited References

Applicant's claimed invention is patentable over MacLeod and Bello, because it includes a combination of limitations not taught or suggested by the cited references, taken individually or in combination. Specifically, neither reference teaches or suggests the steps or elements of the independent claims comprising "generating cardinality estimates for one or more query execution plans for the query using statistics of one or more automatic summary tables that vertically overlap the query," and "using the cardinality estimates to determine an optimal query execution plan for the query."

Nonetheless, MacLeod is cited by the Office Action as teaching all of the steps or elements of the independent claims 1, 11 and 21, except for the use of automatic summary tables (i.e., materialized views), and Bello is cited by the Office Action as teaching the use of automatic summary tables.

Applicant's attorney disagrees.

The portions of MacLeod and Bello cited by the Office Action are set forth below:

> <u>MacLeod: Col. 7, lines 5-65 (emphasis added)</u>
>
> At step 178, the database system implements the execution plan that was selected at step 176. At step 180, the execution plan is stored in an area of memory referred to as the execution plan cache. On subsequent invocations of the same SQL query, rather than retraverse the optimization process, the previously optimized execution plan is retrieved from cache and executed.
>
> Turning now to the present graphical query analyzer, FIG. 5 depicts a user interface in accordance with one embodiment of the present invention, showing graphical analysis of two specified queries. A user at a conventional personal computer 20 specifies via a keyboard 40 a standard SQL query. In this case the query reads:
>
> Select CompanyName, ContactFirstName, ContactLastName
> from Customers
> Where CustomerID=1
>
> Query analyzer interface element (365 on FIG. 1), an application program 36 comprising instructions in memory 22 executed by processing unit 21, causes the query to be displayed in a user interface window 220 shown on a display 47.
>
> The user types the word "go" and the query is thereby submitted to a query analyzer element (370 on FIG. 1), also an application program 36 comprising instructions in memory 22 executed by the processing unit 21. Turning now to FIG. 10, in Step 300 in response to the submission, the query analyzer element 370 performs, or calls upon another application program 36 to perform, the steps shown in FIG. 4 to obtain an execution plan for the submitted query. Again, as discussed in connection with that Figure, **the execution plan is obtained by running a "query optimizer" which selects the most efficient execution plan based on an analysis of database statistics and other information.** However, the execution plan is not implemented or saved as with the database management system described in connection with FIG. 4.
>
> Turning back to FIG. 10, in Steps 310 and 320, the query analyzer element 370 begins a loop for each operator in the execution plan determined in Step 300. The loop is nested within a loop for each statement in a query submitted, which in this case is trivial as only one "Select" statement was submitted
>
> In Step 330, the query analyzer element 370 causes the display of a unique icon corresponding to the operation currently considered in the loop. In particular the query analyzer element 370 may search for a corresponding icon in a table stored on the hard drive 27. The table would include all possible execution operations associated with corresponding icons.

In this step, the query analyzer element 370 also causes the display of the estimated cost, determined during the optimization process (Step 300), for the current operation as a percentage of the cost of the query. A tree branch to a parent operation (e.g., one that called the current operation) is also displayed.

In Step 340, the display step is repeated for each operation in the query execution plan determined in Step 300. In this way an intuitive, graphical analysis of execution of the current query is displayed in the form of a tree structure 210.

Bello: Col. 10, lines 8-45 (emphasis added)

Various criteria may be used during this pruning process. For example, one possible pruning criteria may be that at least one of the tables referenced in the received query must be a base table of the materialized view. Based on this criteria, a materialized view that has base tables A, B, and C would qualify as a "possible materialized view" with respect to a query that requires a join between tables A and D. On the other hand, a materialized view that has base tables B, C and E would not qualify as a "possible materialized view" with respect to a query that requires a join between tables A and D.

Steps 202 and 204 form a loop in which each materialized view in the set of possible materialized views is processed. During the processing of each materialized view, the database server determines whether the materialized view is actually eligible to be used in a rewrite of the received query, and if so, the relative benefit gained by using that particular materialized view. Specifically, at step 202, it is determined whether any "possible materialized views" are left to be processed. If all possible materialized views have been processed, control passes to step 220. Otherwise, control passes to step 204.

At step 204, an unprocessed possible materialized view is selected. At step 206, it is determined whether the selected materialized view is eligible for use in rewriting the received query. If the selected materialized view is not found to be eligible, control passes to step 230, and the materialized view is removed from consideration. From step 230, control passes back to step 202.

If the selected materialized view is found to be eligible at step 206, control passes to step 214. At step 214, a "query reduction factor" is computed for the materialized view currently being processed. The query reduction factor is a value that estimates how useful it will be to access the materialized view to process the received query. The higher the query reduction factor, the greater the estimated benefit of using the materialized view to process the query.

Nothing in the above descriptions from MacLeod and Bello can fairly be said to represent "generating cardinality estimates for one or more query execution plans for the query using statistics of one or more automatic summary tables that vertically overlap the query," and "using the cardinality estimates to determine an optimal query execution plan for the query."

The above portions of MacLeod merely state that an estimated cost for an operation in a query is determined during an optimization process, wherein the estimated cost for the operation is displayed as a percentage of the cost of the query.

Similarly, the above portions of Bello merely state that the database server determines whether the materialized view is actually eligible to be used in a rewrite of the received query, and if so, the relative benefit gained by using that particular materialized view.

However, nowhere does either reference describe the use of cardinality estimates generated using statistics of automatic summary tables that vertically overlap the query. As noted in Appellant's specification, and as acknowledged by the Office Action, automatic summary tables are pre-computed queries or materialized views. Also as noted in Appellant's specification, although not acknowledged by the Office Action, an automatic summary table vertically overlaps a query when the set of predicates applied by the automatic summary table is a subset of the predicates required by the query. However, there is no discussion of vertically overlapping automatic summary tables in either MacLeod or Bello.

Consequently, even when combined, the MacLeod and Bell references do not teach or suggest Applicant's invention. Moreover, the various elements of Applicant's claimed invention together provide operational advantages over the cited references. In addition, Applicant's invention solves problems not recognized by the cited references.

Thus, Applicant submits that independent claims 1, 11 and 21 are allowable over MacLeod and Bello. Further, dependent claims 3-10, 13-20 and 23-30 are submitted to be allowable over MacLeod and Bello in the same manner, because they are dependent on independent claims 1, 11 and 21, respectively, and because they contain all the limitations of the independent claims.

F. <u>Appellant's Dependent Claims Are Patentable Over The Cited References</u>

In addition, Appellant's dependent claims are patentable over MacLeod and Bello, because they recite a combination of limitations not taught or suggested by the cited references, taken individually or in any combination

With regard to claims 3, 13 and 23, which recite that "the statistics of the one or more automatic summary tables are used to improve a combined selectivity estimate of one or more predicates of the query," the Office Action asserts that MacLeod teaches these elements at col. 8, lines 20-54. Appellant's attorney disagrees. At the indicated location, MacLeod merely describes the cost of a Table Scan, a Hash Match and scalar computations, the approximate cost of each query as a percentage of a batch total, and the display of the cost of each operation as a percentage of the total query cost. Nowhere in this portion of MacLeod is there a description of a combined selectivity

estimate of one or more predicates of a query or using the statistics of the automatic summary tables to improve the combined selectivity estimate of the predicates.

With regard to claims 4, 14 and 24, these claims stand or fall with claims 3, 13 and 23.

With regard to claims 5, 15 and 25, which recite that "the selectivity estimate comprises a ratio of a cardinality of the automatic summary table to a product of cardinalities of base tables referenced in the automatic summary table and the query," the Office Action asserts that MacLeod teaches these elements at col. 6, lines 10-61. Appellant's attorney disagrees. At the indicated location, MacLeod merely describes the use of database statistics in identifying execution plans that require processing of minimal numbers of records, wherein the database statistics typically include: the time of the last statistics collection; the number of rows in the index; the number of pages occupied by the index; the average row length; the distribution of values in the indexed column (i.e. a histogram); the densities of values in the indexed column; and the number of rows used to produce the histogram and density information. Nowhere in this portion of MacLeod is there a description of a selectivity estimate that comprises a ratio of a cardinality of the automatic summary table to a product of cardinalities of base tables referenced in the automatic summary table and the query.

With regard to claims 6, 16 and 26, these claims stand or fall with claims 3, 13 and 23.

With regard to claims 7, 17 and 27, these claims stand or fall with claims 6, 16 and 26.

With regard to claims 8, 18 and 28, which recite "determining a subpredicate combined selectivity estimate of the unapplied eligible predicates using column distribution statistics of the automatic summary table," the Office Action asserts that Bello teaches these elements at col. 10, lines 30-36. Appellant's attorney disagrees. The indicated location in Bello merely describes determining whether a previously unselected materialized view is eligible for use in rewriting the received query. Nowhere in this portion of Bello is there a description of determining a subpredicate combined selectivity estimate of the unapplied eligible predicates using column distribution statistics of the automatic summary table.

With regard to claims 9, 19 and 29, which recite that "a cardinality ratio comprises a ratio of a cardinality of the automatic summary table to a product of cardinalities of base tables referenced in the automatic summary table and the query," the Office Action asserts that MacLeod teaches these elements at col. 8, lines 20-55. Appellant's attorney disagrees. At the indicated location, MacLeod describes displaying a tree structure showing the cost of each operation of the query as a percentage of the total query cost, as well as showing the approximate cost of each query as a percentage of a batch total. Nowhere in this portion of MacLeod is there a description of a cardinality ratio

comprises a ratio of a cardinality of the automatic summary table to a product of cardinalities of base tables referenced in the automatic summary table and the query.

With regard to claims 10, 20 and 30, which recite that "the selectivity estimate comprises a product of the subpredicate combined selectivity estimate and the cardinality ratio," the Office Action asserts that MacLeod teaches these elements at col. 7, lines 55-60. Appellant's attorney disagrees. At the indicated location, MacLeod merely describes the query analyzer element displaying the estimated cost, determined during the optimization process, for the current operation as a percentage of the cost of the query. Nowhere in this portion of MacLeod is there a description of a selectivity estimate comprising a product of the subpredicate combined selectivity estimate and a cardinality ratio.

IV. Conclusion

In view of the above, it is submitted that this application is now in good order for allowance and such allowance is respectfully solicited. Should the Examiner believe minor matters still remain that can be resolved in a telephone interview, the Examiner is urged to call Applicant's undersigned attorney.

Respectfully submitted,

GATES & COOPER LLP
Attorneys for Applicants

Howard Hughes Center
6701 Center Drive West, Suite 1050
Los Angeles, California 90045
(310) 641-8797

Date: April 4, 2006

By: ____________________
Name: George H. Gates
Reg. No.: 33,500

GHG/

G&C 30571.272-US-U1